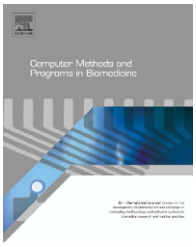
COMPUTER METHODS AND PROGRAMS IN BIOMEDICINE 109 (2013) 104–111

journal homepage: www.intl.elsevierhealth.com/journals/cmpb



KmL3D: A non-parametric algorithm for clustering joint trajectories

C. Genolini[a,b,*], J.B. Pingault[c], T. Driss[b], S. Côté[c,d,e], R.E. Tremblay[c,d,e,g], F. Vitaro[c,d], C. Arnaud[a], B. Falissard[e,f]

[a] U1027, INSERM, Université Paul Sabatier, Toulouse III, France
[b] CeRSM (EA 2931), UFR STAPS, Université de Paris Ouest-Nanterre-La Défense, France
[c] Research Unit on Children's Psychosocial Maladjustment, University of Montreal and Sainte–Justine Hospital, Montreal, Quebec, Canada
[d] International Laboratory for Child and Adolescent Mental Health Development, University of Montreal, Montreal, Quebec, Canada
[e] INSERM U669, Paris, France
[f] University Paris–Sud and University Descartes, Paris, France
[g] School of Public Health, Physiotherapy and Population Science, University College Dublin, Dublin, Ireland

ARTICLE INFO

Article history:
Received 8 December 2011
Received in revised form
20 August 2012
Accepted 23 August 2012

Keywords:
Longitudinal data
k-means
Cluster analysis
Non-parametric algorithm
Joint trajectories

ABSTRACT

In cohort studies, variables are measured repeatedly and can be considered as trajectories. A classic way to work with trajectories is to cluster them in order to detect the existence of homogeneous patterns of evolution.

Since cohort studies usually measure a large number of variables, it might be interesting to study the joint evolution of several variables (also called joint-variable trajectories). To date, the only way to cluster joint-trajectories is to cluster each trajectory independently, then to cross the partitions obtained. This approach is unsatisfactory because it does not take into account a possible co-evolution of variable-trajectories.

KmL3D is an R package that implements a version of k-means dedicated to clustering joint-trajectories. It provides facilities for the management of missing values, offers several quality criteria and its graphic interface helps the user to select the best partition. KmL3D can work with any number of joint-variable trajectories. In the restricted case of two joint trajectories, it proposes 3D tools to visualize the partitioning and then export 3D dynamic rotating-graphs to PDF format.

© 2012 Elsevier Ireland Ltd. All rights reserved.

1. Introduction

A cohort study is a longitudinal study where variables are measured repeatedly over time. For each patient, these variables evolve over time; they will be referred as the "variable-trajectory". A standard way to work with variable-trajectories is to cluster them in order to detect the existence of homogeneous patient subgroups. Many methods have been developed for this purpose [1–5]. All these methods cluster according to a single variable-trajectory.

Since cohort studies usually measure a large number of variables, it might be interesting to study the joint evolution of several variable-trajectories (also called joint-trajectories). To date, this has not been possible: the only way to cluster joint-trajectories is to cluster each variable-trajectory independently, then to consider the combination of the partitions obtained. In the case of two variable-trajectories (A) and (B)

* *Corresponding author at*: U1027, INSERM, Université Paul Sabatier, Toulouse III, France. Tel.: +33 1 58 41 28 52; fax: +33 1 58 41 28 43.
E-mail address: genolini@u-paris10.fr (C. Genolini).
0169-2607/$ – see front matter © 2012 Elsevier Ireland Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.cmpb.2012.08.016

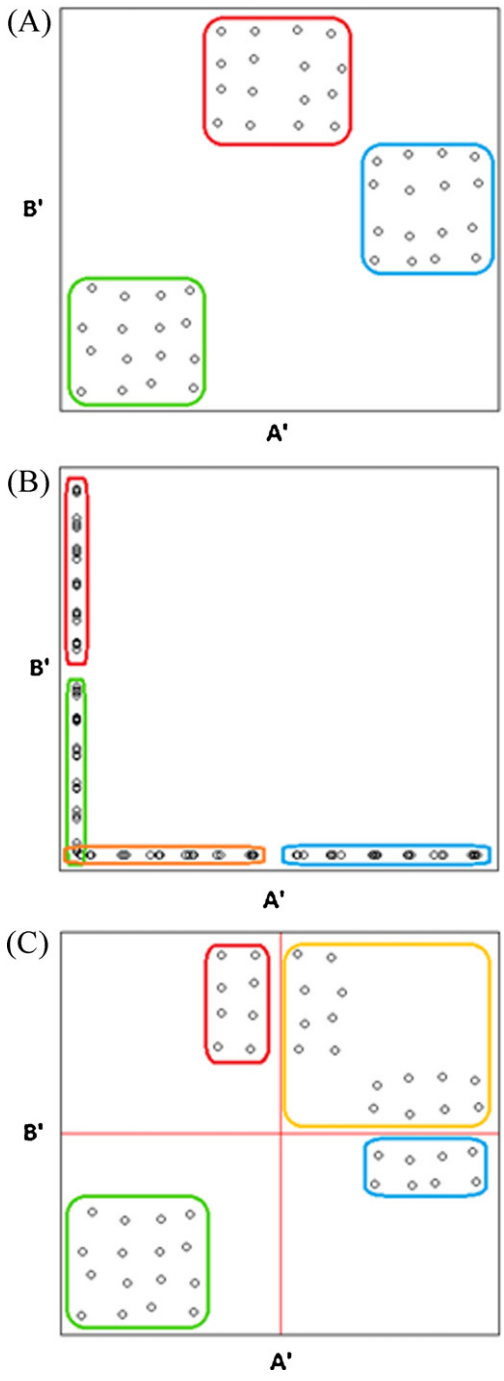


Fig. 1 – Clustering two variables jointly (A) or using the cross-partition (C). (A) Clustering considering A′ and B′ jointly. (B) Clustering A′ and B′ separately. (C) Cross-partition using the clusters found in B. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of the article.)

that need to be considered simultaneously, authors determine a partition $P^{(A)}$ by clustering only (A) and then they determine a partition $P^{(B)}$ by clustering only (B). Finally, according to their needs, they either use $P^{(A)}$ and $P^{(B)}$, or the cross-partition $P^{(A\times B)} = P^{(A)} \times P^{(B)}$. This approach is of limited value for two reasons. One advantage of classification methods is to enable the conversion of continuous data into categorical data, after which the categories obtained can be used, for instance in a regression model. If the two variables (A) and (B) are linked in some way, partitions $P^{(A)}$ and $P^{(B)}$ will be correlated. So the inclusion of $P^{(A)}$ and $P^{(B)}$ in the same regression will lead to instability of the model. Another weakness of the method is that partition $P^{(A\times B)}$ does not enable detection of groups where the co-evolution of the two variables is complex. By analogy, consider two classic variables[1] A′ and B′ ('classic variable' opposed to 'variable-trajectories') plotted in Fig. 1A. There are clearly three clusters. If we cluster according to variable A′ and then according to variable B′, we identify two groups for A′ (Fig. 1B, orange and blue) and two groups for B′ (Fig. 1B, green and red). The cross-partition resulting from these two clustering procedures is presented Fig. 1C. Four groups are obtained, but not those found by clustering the two variables jointly (Fig. 1A). This example underlines the need for a clustering method that considers several variable-trajectories simultaneously.

In addition, clustering several continuous correlated variables (the trajectories) in a single nominal variable (groups) summarize information of correlated variable. This makes the use of this information for further statistical analysis much easier. For example, a single nominal variable can be used in a regression (as we show in example "inattention", Section 2.3) whereas the inclusion of joint trajectories in such model would not have been possible.

kml3d, from the package KmL3D [6], is a partitioning algorithm that works jointly on several variable trajectories. It is based on the k-means algorithm [7,8]. It has the same advantages as KmL (management of missing values, several quality criteria, graphic interface to select the best partition [9,10]). It also provides 3D tools for visualizing the partitioning of the joint-trajectories or for exporting 3D rotating-graphs to PDF format [11].

The rest of this paper is organized as follows: Section 2 presents KmL3D, a new implementation of k-means designed to cluster joint trajectories. Section 3 contains simulations on both artificial and real data. The performances of KmL3D are compared to the results obtained using classic clustering on each variable, then considering the cross-partition. Section 4 is the discussion.

2. Materials and methods

2.1. Algorithm

2.1.1. Notations

Let S be a set of n subjects. For each subject, m outcome variables $Y_{..A}, Y_{..B}, \ldots, Y_{..M}$ at t different times are measured. $Y_{..A}$ is called a *single variable-trajectory* (or *variable-trajectory*). Several variable-trajectories ($Y_{..A}, Y_{..B}, \ldots, Y_{..M}$) considered jointly are called *joint variable-trajectories*.

For subject i, the value of $Y_{..A}$ at time j is noted y_{ijA}. The sequence $y_{i.A} = (y_{i1A}, y_{i2A}, \ldots, y_{itA})$ is called a single

[1] We present here an example using classic variables for the convenience of graphic representation, but all this is directly transposable to variable-trajectories.

trajectory[2] (or trajectory). Several single trajectories $y_{i..} = \begin{pmatrix} y_{i.A} \\ y_{i.B} \\ \dots \\ y_{i.M} \end{pmatrix}$ are called joint trajectories. Overall, $y_{i..}$ is a matrix $y_{i..} = \begin{pmatrix} y_{i1A} & y_{i2A} & \cdots & y_{itA} \\ y_{i1B} & y_{i2B} & & y_{itB} \\ & \vdots & \ddots & \vdots \\ y_{i1M} & y_{i2M} & \cdots & y_{itM} \end{pmatrix}$ where lines are single variable trajectories. If j is fixed, the sequence $y_{ij.} = \begin{pmatrix} y_{ijA} \\ y_{ijB} \\ \dots \\ y_{ijM} \end{pmatrix}$ is called individual's state at time j. The individual's state at time j is the jth column of the matrix $y_{i..}$.

The aim of clustering is to divide S into k homogeneous sub-groups.

2.1.2. *k-means*

k-means is a non-parametric hill-climbing algorithm [12] belonging to the EM class (Expectation–Maximization) [13]. It works as follows: initially, each observation is assigned to a cluster. Then the optimal clustering is reached by alternating two phases. During the *Expectation* phase, the centers of each cluster are computed. The *Maximization* phase then consists in assigning each observation to its "nearest cluster". The alternation of the two phases is repeated until no further changes occur in the clusters. k-means is non-parametric in the sense that there is no need to make hypothesis neither on the variables distribution nor on the shape of the means trajectories of each groups.

In the case of longitudinal data, "cluster centers" are the mean trajectory of each group, that is to say the mean of all the individual trajectories that belong to the clusters. For an individual i, the "nearest cluster" C is the cluster that minimizes the distance between i and the mean trajectory of C. This concept is strongly related to the concept of distance, which we will now define.

2.1.3. *Distance*

k-means can work with various distances: Euclidean, Manhattan, Minkowski (the generalization of the two previous distances) and many others. Working on joint-trajectories raises the question of the distance between two joint-trajectories. More precisely, considering the joint-trajectories of two individuals $y_{1..}$ and $y_{2..}$, we seek to define $d(y_{1..}, y_{2..})$, the distance between $y_{1..}$ and $y_{2..}$. Strictly speaking, this is the distance between two matrices. Several methods are possible, we will focus on two. The first is to consider the t columns of the two matrixes, then compute t distances between the t couples of columns and finally to combine these t distances using a function that will combine the "column-distances". The second is to consider the m lines of the two matrixes, then compute m distances between the m couples of lines and finally to combine these m distances using a function that will combine the "line-distances".

More formally, let *Dist* be a distance function and $||\cdot\cdot||$ be a norm. To compute a distance d between $y_{1..}$ and $y_{2..}$ according to the first method, for each fixed j, we define the distance between $y_{1j.}$ and $y_{2j.}$ (distance between the individuals' state at time j) as $d_{j.}(y_{1j.}, y_{2j.}) = Dist(y_{1j.}, y_{2j.})$. This is the distance between column j in matrix $y_{1..}$ and column j in matrix $y_{2..}$. The result is a "vector of t distances" $(d_{1.}(y_{11.}, y_{21.}), d_{2.}(y_{12.}, y_{22.}), \dots, d_{t}(y_{1t.}, y_{2t.}))$. Then we combine these t distances using a function that algebraically corresponds to a norm $|||\cdots||$ of the vector of distance. Overall, the distance between $y_{1..}$ and $y_{2..}$ is $d(y_{1..}, y_{2..}) = ||\ (d_{1.}(y_{11.}, y_{21.}), d_{2.}(y_{12.}, y_{22.}), \dots, d_{t.}(y_{1t.}, y_{2t.}))\ ||$.

To compute a distance d' between $y_{1..}$ and $y_{2..}$ according to the second method, for each variable X, we define the distance between $y_{1.X}$ and $y_{2.X}$ (distance between the individual trajectories X) as $d_{.X}(y_{1.X}, y_{2.X}) = Dist(y_{1.X}, y_{2.X})$. This is the distance between line X in matrix $y_{1..}$ and line X in matrix $y_{2..}$.The result is a "vector of m distances" $(d_{.A}(y_{1.A}, y_{2.A}), d_{.B}(y_{1.B}, y_{2.B}), \dots, d_{.M}(y_{1.M.}, y_{2.M}))$. Then we combine these m distances by considering the norm $||\cdots||$ of the vector of distance. Overall, $d'(y_{1..}, y_{2..}) = ||(d_{.A}(y_{1.A}, y_{2.A}), d_{.B}(y_{1.B}, y_{2.B}), \dots, d_{.M}(y_{1.M.}, y_{2.M}))\ ||$.

The choice of the norm $||\cdot\cdot||$ the distance *Dist* and method d or d' can lead to the definition of a large number of distances between $y_{1..}$ and $y_{2..}$. In practice, the standard p-norm for $||\cdot\cdot||$ and the Minkovsky distance with parameters p for *Dist* give the same result: $d(y_{1..}, y_{2..}) = d'(y_{1..}, y_{2..})$

Proof.

$$\begin{aligned} d(y_{1..}, y_{2..}) &= \sqrt[p]{\sum_j (d_{j.}(y_{1j.}, y_{2j.}))} \\ &= \sqrt[p]{\sum_j \left(\sqrt[p]{\sum_X |y_{1jX} - y_{2jX}|^p} \right)^p} \\ &= \sqrt[p]{\sum_j \sum_X |y_{1jX} - y_{2jX}|^p} \\ &= \sqrt[p]{\sum_X \left(\sqrt[p]{\sum_j |y_{1jX} - y_{2jX}|^p} \right)^p} \\ &= \sqrt[p]{\sum_X (d_{.X}(y_{1.X}, y_{2.X}))^p} \\ &= d'(y_{1..}, y_{2..}) \end{aligned} \tag{1}$$

We can therefore define the Minkowski distance between two joint variable trajectories:

$$Dist(y_{1..}, y_{2..}) = \sqrt[p]{\sum_j \sum_X |y_{1jX} - y_{2jX}|^p} \tag{2}$$

The Euclidean distance is obtained by setting $p = 2$, the Manhattan distance by setting $p = 1$ and the maximum distance by passing to the limit $p \to +\infty$. In practice, KmL3D uses Euclidean distance as the default distance. But it also allows users to define their own distance.

2.1.4. *Standardization*

Since cohort studies deal with several different kinds of variables, the joint variables cannot be measured on the same scale. This problem has already been extensively discussed in the classic (non-trajectory) situation [12]. A possible solution is then to normalize the data. This can also be done with trajectories. KmL3D provides functions to normalize the

[2] Strictly speaking, it should be called *single individual trajectory*. But the current practice is to omit the word "individual".

variable-trajectories. A small difference with the classic situation exists, as each variable-trajectory is not normalized at each time but in its entirety: let $\overline{Y^{(A)}}$ and $sd^{(A)}$ be respectively the mean and the standard deviation of all $y_{ij}^{(A)}$ (for each i and j). Then the outcome $y_{ij}^{(A)}$ becomes:

$$y'^{(A)}_{ij} = \frac{(y_{ij}^{(A)} - \overline{y^{(A)}})}{sd^{(A)}} \quad (3)$$

The normalized joint trajectory $y'_{i..}$ is obtained by normalization of its single trajectories $y'^{(X)}_{i.}$ one by one.

2.1.5. *Visualization*

The partitioning of longitudinal data allows the identification of homogeneous subgroups. One of the advantages of this technique is to exhibit the average trajectory of each groups. These mean trajectories summarize the overall evolution of the group, thus highlighting specific behaviors. The obtained clusters can then be used in statistical analyses, either as an explanatory or as a dependent variable. It is therefore important to be able to graphically display these typical trajectories. Working on single-trajectories, the plot is fairly simple: let us consider a coordinate system (O,x,y). The time is placed on the axis of abscissa [O,x), the variable is on the vertical axis [O,y).

Drawing joint trajectories is more complex. A graphic representation is possible in the case of two joint-trajectories, by using a three-dimensional coordinate system (O,x,y,z): time is on axis [Ox), the first variable is on axis [O, y), the third is on axis [O, z). This gives a 3D representation of the evolution of the joint-trajectories (which explains the name of the package KmL3D). It is interesting to note that recent developments in pdf format let the user include 3D dynamic graphs in pdf documents. The user can rotate the graph, changing the point of view, with the mouse. This can be very convenient to provide scope for displaying joint-trajectories in Scientific articles. Examples of this kind of graph are presented Figs. 2a–c and 3.

2.1.6. *Optimal number of clusters, dealing with missing data, avoiding local maximum*

The choice of the optimum number of clusters is based on the Calinski and Harabatz criterion:

$$c(k) = \frac{\text{Trace}(B)}{\text{Trace}(W)} \frac{n-k}{k-1} \quad (4)$$

where B is the matrix of variance between, W the matrix of variance within, n the number of individuals and k the number of groups (see Ref. [14] for details). Since the limits of this type of quality criterion are well known [15], two other criteria are also available: Ray and Turi [16] and Davies and Bouldin [17]. The Ray and Turi criterion is:

$$r(k) = \frac{\text{DW}}{\text{DB}} \quad (5)$$

where DW, the distance within, is Σ_iDistance(i, center(i)) and DB, the distance between is $\text{DB} = \min_{i \neq j}(\text{Distance}(\text{center}(i), \text{center}(j)))$.

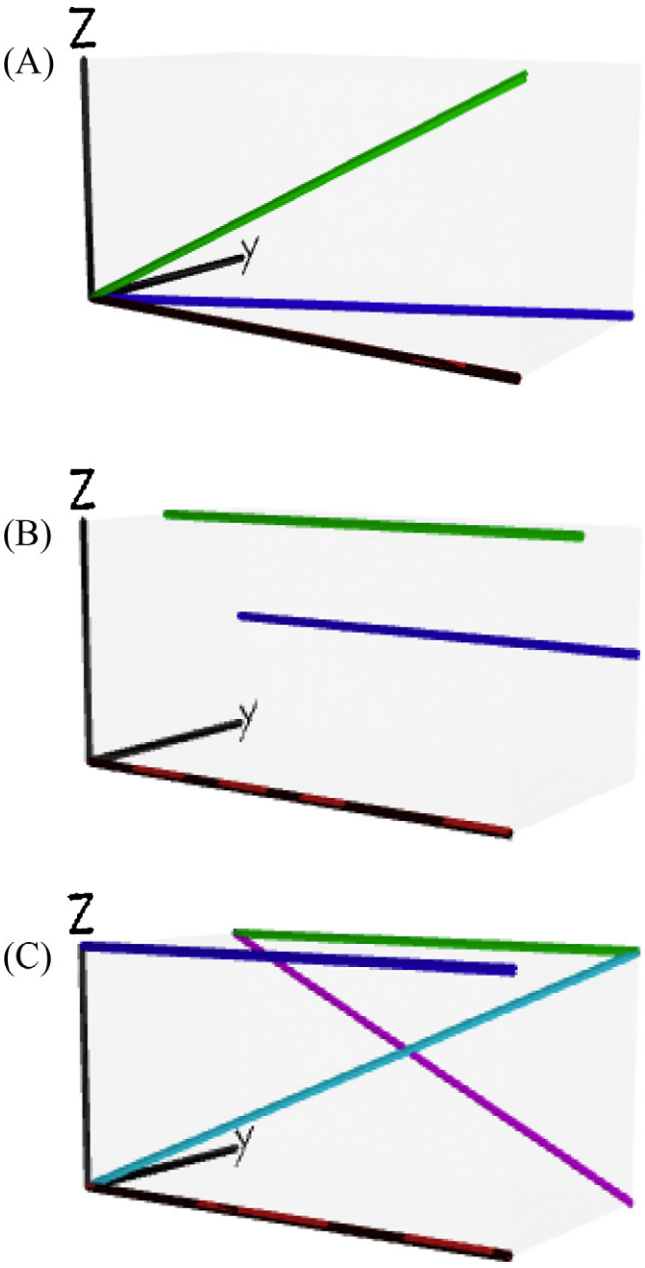


Fig. 2 – Different data set shapes used for generating artificial data. (A) Dataset "Three diverging lines". (B) Dataset "Three parallel lines". (C) Dataset "Five lines".

Davies and Bouldin criterion is:

$$d(k) = \text{Mean}(\text{Proximity}(\text{cluster}(i), \text{clusters}(j))) \quad (6)$$

where $\text{Proximity}(i, j) = (\text{DistInternal}(i) + \text{DistInternal}(j))/(\text{DistExternal}(i, j))$. The definition of DistInternal and DistExternal can lead to various measures. In KmL3D, we use the classic distance "average to the center" for DistInternal and "distance between centers" for DistExternal.

In addition, a graphic interface enables the user to visualize the partition obtained.

The management of missing data [18–20] is performed either by imputing the trajectories or by using distances with

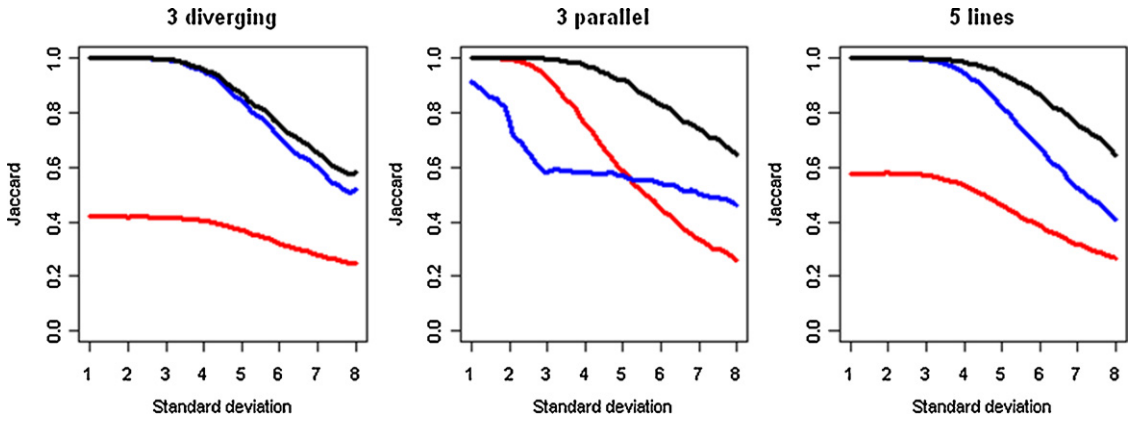


Fig. 3 – Effect of standard deviation noise using the Jaccard similarity index, according to the shape (black: P_{3D}/blue: $P^{(A \times B)}$/red: $P^{(A \times B\text{-max})}$). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of the article.)

Gower adjustment [12]. Available imputing methods include LOCF (Last Occurrence Carried Forward: a missing value is allocated the previous known value), linear interpolation (a line is drawn between the known values surrounding the missing ones) or CopyMean (imputation needs two steps: first, linear interpolation is used; then a variation *copying* the population mean trajectory is added. For more details, see Ref. [10]).

2.2. *Simulation data*

Conventional clustering techniques involve partitioning the variables one after the other, then considering the cross-partitions. KmL3D makes it possible to cluster joint-trajectories. In order to compare the efficiency of the two approaches, we compared the procedures on both simulated and real data. For simplicity, we worked on two variable-trajectories, but kml3d can cater for more.

We worked on 4200 data sets defined as follows: A *data set shape* is defined by a number of groups and, for each groups, two real functions (from **R** to **R**). These two functions define the typical joint trajectory that follows individuals in the group. Jointly, they can be considered as a function from **R** to $\mathbf{R}^2$ which is called the *theoretical joint trajectory*. For example, if we study the joint evolution of night sleep duration and hyperactivity, the first function is one that associates sleep duration to each time point, and the second associates a score for hyperactivity to each time. Together, these two functions define the joint-trajectory that associate a pair of data (sleep duration, hyperactivity) to each time. A data set shape is a given number of groups and a theoretical joint trajectory for each group. For our simulations, we defined 3 data set shapes (Fig. 2A–C):

- 1: In "Three diverging lines", there are three groups A, B and C. The theoretical joint trajectories are: $f_A(k) = (0,0)$; $f_B(k) = (0,k)$; $f_C(k) = (k,0)$ with k in [0:10].
- 2: In "Three parallels lines", there are three groups A, B and C. The theoretical joint trajectories are: $f_A(k) = (0,0)$; $f_B(k) = (4,8)$; $f_C(k) = (8,4)$ with k in [0:10].
- 3: In "Fives lines", there are five groups A, B, C, D and E. The theoretical joint trajectories are: $f_A(k) = (0,0)$; $f_B(k) = (10,10)$; $f_C(k) = (0,10)$; $f_D(k) = (k,k)$; $f_E(k) = (10,10-k)$ with k in [0:10].

Data sets are then created from the data set shape. Initially, a number of individuals per group is set (either 50 or 200). The trajectory of an individual is obtained by adding a residual variation to the theoretical joint trajectory of his group. Individual variations randomly follow a normal distribution with mean (0, 0) and variance (σ^2, σ^2). The standard deviation varies from 1 to 8 by steps of 0.01. Since the distance between two theoretical joint trajectories is around 10, $\sigma = 1$ provides some "easily identifiable and distinct clusters" whereas $\sigma = 8$ gives some "very overlapping groups". Overall, 3 (shapes) times 2 (number of subjects) times 700 (standard deviation) give 4200 data sets.

For each data set, we clustered the joint-trajectories using KmL3D. This partition called *joint partition* is noted P_{3D}. Then we constructed two univariates partitions considering each variable-trajectory separately (using kml from package KmL [21]). These two partitions are noted $P^{(A)}$ and $P^{(B)}$. Finally, $P^{(A \times B)}$ is obtained by crossing the two partitions $P^{(A)}$ and $P^{(B)}$. The partition obtained is called the *cross univariate partition*. Note that the number of clusters found by kml is not necessarily the true number of clusters. For example, on the data shape "five lines", the projection of population (A) is partitioned into four groups whereas the projection of population (B) is partitioned into three groups. So we also constructed partitions $P^{(A\text{-max})}$ and $P^{(B\text{-max})}$ based on the real number of clusters present in the artificial data set (5 for the shape "five lines", 3 for the two other shapes). Then $P^{(A \times B\text{-max})}$ is the partition obtained by crossing $P^{(A\text{-max})}$ and $P^{(B\text{-max})}$. It is called the *maximum cross partition* (maximum referring to the number of clusters). This partition may seem irrelevant for the detection of clusters and present quite a large number of clusters (25 in the case of "five lines"), but because it is the current method used in the processing of joint trajectories, it is important to consider its performances.

To check the quality of the procedure, we compared the partition it found with the true partition, P_{TRUE} (on artificial

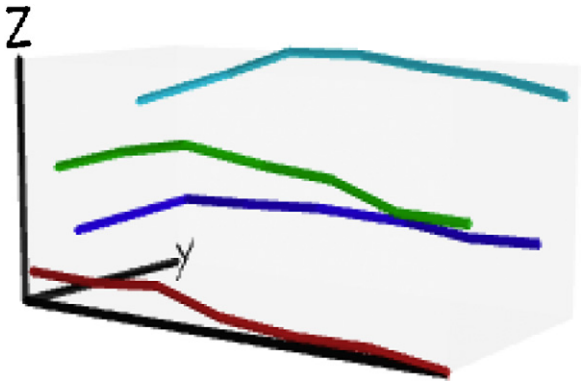


Fig. 4 – Joint trajectories of inattention evaluate by the mother (Y) or the teacher (Z).

data, P_{TRUE} is known). The closer a partition is to P_{TRUE}, the better is its quality. The similarity index that assesses the proximity between the partition and P_{TRUE} is the Jaccard index [22,23].

2.3. *Real data*

Our first real example is derived from Ref. [24]. The objective of the study was to determine the link between inattention and high school graduation, as inattention was shown to be predictive of educational attainment [25,26]. The participants consisted of 2000 children (1001 boys) randomly selected from respondents in a larger representative sample of kindergarten children from the province of Quebec, in 1986–1987. Children were rated on one side by the teacher and on the other side by the mother. The rating was performed using the Social Behavior Questionnaire (SBQ) [27] each year between kindergarten and sixth grade either by teachers or by mothers, which provided seven assessments between the ages of 6 and 12 years. These two assessments were highly correlated so they could not both be used as predictors for high school graduation.

Three partitions can be computed: the first (Mr) uses mother ratings alone (by KmL), the second (Tr) is computed using teacher ratings alone (by KmL), the third (MTr) uses dimensional trajectories with assessments from both informants (by KmL3D, see Fig. 4). All three partitions present four groups. Since the different quality criteria all disagree on the number of groups, the choice was based on the literature and according to expert advice. They are close to each other, but there are some differences. They give the estimations of three models that are presented in Table 2. As expected from the literature, teacher assessments of inattention were clearly more predictive than maternal assessments. Taking into account both informants improved the fit of the model as shown by the decrease in the Akaike Information Criterion (AIC). Furthermore, a higher pseudo R-squared indicates which model better predicts the outcome (when compared to a pseudo R-squared on the same data, predicting the same outcome, which is the case here). In the present case, model 3 with three dimensional trajectories based on both informants had the highest pseudo-R^2.

Table 1 – Average Jaccard similarity index, according to the shape.

Dataset shape	P_{3D}	$P^{(A \times B)}$	$P^{(A \times B\text{-max})}$
3 Diverging	0.86	0.83	0.36
3 Parallel	0.90	0.61	0.67
5 Lines	0.91	0.81	0.47

2.4. *Real data "sleep duration"*

Our second example is derived from Touchette et al. [28]. In a sample of 2057 children aged 1.5–5 years, night-time sleep duration and hyperactivity were measured yearly by questionnaires administered to mothers. The aim of the study was to investigate the developmental trajectories in relation to night-time sleep duration and hyperactivity over the preschool years.

3. Results

3.1. *Simulated data*

Table 1 and Fig. 3 show the results. The example "three diverging" represents trajectories for which the co-evolution corresponds to a simple crossover of the two variable-trajectories. Not surprisingly, both methods *joint partition* P_{3D} and *cross partition* $P^{(A \times B)}$ give good results. *Maximum cross partition* $P^{(A \times B\text{-max})}$ gives the worst results. "Three parallel" presents an illusion of simplicity. In practice, the noise added to the trajectories makes it difficult to reconstruct the clusters when each variable is considered independently. The *joint partition* exhibits good performances (fairly close to those obtained on "Three diverging lines"). The *cross partition* and *maximum cross partition* gives less good results. "Five lines" is an example representing a more complex co-evolution of the two variables. Once again, *joint partition* gives good results, *cross partition* is not as good and *maximum cross partition* is the worst.

3.2. *Real data "inattention"*

All three partitions present four groups. They are close to each other, but some differences exist. They lead to the estimation of three models that are presented Table 2. Group 'Inattention A' is the reference. As expected from the literature, teacher assessments of inattention were clearly more predictive than maternal assessments. Taking into account both informants improved the prediction with higher odds ratios, in particular for the two highest trajectories, as well as better fit statistics (including the pseudo-R^2).

3.3. *Real data "sleep duration"*

Using KmL on the single trajectory variables gives results very close to those computed using Proc Traj published in the article. Conversely, the analysis of the joint trajectories with KmL3D gives very different results from those obtained by crossing the single variable partitions. If we compare the crossed partition with the kml3d partition, we find that only 948 individuals (out of 1917) are classified in

Table 2 – Prediction of high school graduation failure according to inattention.

	Mother ratings (Mr)			Teacher ratings (Tr)			Mother and teacher ratings (MTr)		
	%	aRR	95% CI	%	aRR	95% CI	%	aRR	95% CI
Inattention A	12.9%	-	-	9.9%	-	-	8.4%	-	-
Inattention B	24.1%	1.67***	1.30–2.14	32.9%	2.85***	2.22–3.67	23.5%	2.52***	1.89–3.37
Inattention C	43.5%	2.81***	2.22–3.55	45.4%	3.86***	3.05–4.88	48.8%	4.74***	3.64–6.17
Inattention D	63.6%	3.73***	2.93–4.74	69.2%	5.52***	4.40–6.91	67.8%	6.48***	5.01–8.37
AIC	1888.9	1752.7	1714.2						
Nagelkerke Pseudo-R^2	0.27	0.35	0.38						

Note: Percentages of participants failing to graduate in each trajectory are presented the first column of each model (%). Odds ratios tend to overestimate the risk for common outcomes [29] which is the case here; we present risk ratios instead. Unadjusted risk ratios can be calculated by simply dividing the percentages in each trajectory (e.g. for mother ratings, the risk ratio for not graduating when comparing participants in the high [D] and the low reference group [A] is 63.6/12.9 = 4.9). Adjusted risk ratios were estimated by fitting a Poisson regression to the binary outcome, using a sandwich estimator of variance to estimate confidence intervals (95% CI).

*** <.001.

the same way, which is slightly less than 50%. The Jaccard similarity index between the two partitions is 0.25. In addition, KmL3D suggests the use of 4 groups instead of 12, in which individuals would be divided as follows: A: 28.8% [hyper = medium high/sleep = high]; B: 28.2% [hyper = medium low/sleep = low]; C: 27.5 [hyper = low/sleep = high]; D: 15.4% [hyper = high/sleep = low].

4. Discussion

This article presents the package KmL3D, a version of k-means adapted to the analysis of joint-trajectories. Our package works on R platform and is available at Ref. [6]. Like KmL, it is able to deal with missing values, it provides an easy way to run the algorithm several times and its graphical interface helps the user to choose the appropriate number of clusters when criteria traditionally devoted to this task are not efficient. It also provides 3D devices for displaying and exporting results. Clustering trajectories is a major issue for statistical analysis of cohort studies. It makes it possible to define the typical trajectories that follow a population (for instance typical trajectories for inattention). It also summarize several continuous correlated variables (the trajectories) in a single nominal variable (groups) which are easier to use for further statistical analysis (such as modeling high school graduation using typical groups for inattention).

The classic approach to cluster joint-trajectories is to find a univariate partition for each variable-trajectory, then to consider the cross univariate partition. KmL3D clusters data taking into account the co-evolution of the trajectories. To evaluate the effectiveness of the method, we compared it to the classic approach. On artificial data (data for which the group structure is known), the performance of kml3d was clearly better than conventional techniques.

These performances can be partly explained by the fact that the groups in our 3D examples were quite different from those obtained by crossing univariate partitions. It is thus reasonable to ask whether such situations exist in reality. A real example shows that indeed, partitions obtained with kml3d differ from the crossed univariate partitions.

We also explored the impact of the method on the inclusion of groups in a regression. For this, we used real data. The results show that taking into account the co-evolution of variable trajectories improves the understanding of the development of a behavior as well as giving a more accurate estimation of its predictive value.

4.1. *Limitations*

The limitations of KmL3D are inherent to all clustering algorithms. These techniques are mainly exploratory; they cannot statistically test the reality of cluster existence. Moreover, the determination of the optimal cluster number is still an unsettled issue. The EM-algorithm can also be particularly sensitive to the problem of local maximum. KmL3D provides some tools to "see" the joint trajectories in 3D, but these tools can only display 2 variables at the same time. This might be a problem for clustering data using more than two joint variables. Finally, KmL3D is not model-based, which can be an advantage (non parametric, more flexible) but also a disadvantage (no scope for testing goodness of fit).

4.2. *Advantages*

KmL3D provides a way to cluster data according to several joint trajectories. This can help to highlight relationships between complex–variable-trajectories. It could also enable the combining of information available into two strongly-correlated variable-trajectories.

In addition, KmL3D also inherits all the improvements of KmL: as a non-parametric algorithm, it does not need any prior information and it avoids the issues related to model selection. It enables the clustering of trajectories that do not follow polynomial trajectories.

4.3. *Perspectives*

A number of unsolved problems need investigation. The optimization of cluster number, a long-standing and important question, is becoming a more and more crucial issue since it is not possible to graphically represent the result of the partitioning process. Perhaps the particular situation of joint longitudinal data could lead to an efficient solution not yet found in the general context of cluster analysis. Another interesting approach would be to cluster trajectories (still in a

non-parametric manner) with adjustment on covariates. This would reduce the overall variance and thus makes cluster detection more efficient.

REFERENCES

[1] T. Tarpey, K.K.J. Kinateder, Clustering functional data, Journal of classification 20 (1) (2003) 93–114.
[2] F. Rossi, B. Conan-Guez, A. El Golli, Clustering functional data with the SOM algorithm, in: Proceedings of ESANN, 2004, pp. 305–312.
[3] C. Abraham, P.A. Cornillon, E. Matzner-Lober, N. Molinari, Unsupervised curve clustering using B-splines, Scandinavian Journal of Statistics 30 (3) (2003) 581–595.
[4] G.M. James, C.A. Sugar, Clustering for sparsely sampled functional data, Journal of the American Statistical Association 98 (462) (2003) 397–408.
[5] D. Nagin, Group-Based Modeling of Development, Harvard University Press, Cambridge, Massachusetts/London, England, 2005.
[6] C.M. Genolini, kml3d: K-means for joint trajectories, R package version 0.6, http://CRAN.R-project.org/package=kml3d
[7] J. MacQueen, Some methods for classification and analysis of multivariate observations, in: Proceedings of the Fifth Berkeley Symposium on Mathematics, Statistics and Probability, vol. 1, 1966, pp. 281–296.
[8] J.A. Hartigan, M.A. Wong, A K-means clustering algorithm, Journal of the Royal Statistical Society 28 (1979) 100–108.
[9] C.M. Genolini, B. Falissard, KmL: a package to cluster longitudinal data, Computer Methods and Programs in Biomedicine 104 (3) (2011) 112–121.
[10] C.M. Genolini, B. Falissard, KmL: k-means for longitudinal data, Computational Statistics 25 (2) (2010) 317–332.
[11] D. Feng, L. Tierney, Computing and displaying isosurfaces in R, Journal of Statistical Software 28 (1) (2008) 1–24.
[12] B.S. Everitt, S. Landau, M. Leese, in: A. Hodder (Ed.), Cluster Analysis, Arnold Publication, London, 2001.
[13] G. Celeux, G. Govaert, A classification EM algorithm for clustering and two stochastic versions, Computational Statistics and Data Analysis 14 (3) (1992) 315–332.
[14] T. Calinski, J. Harabasz, A dendrite method for cluster analysis, Communications in Statistics 3 (1) (1974) 1–27.
[15] Y. Shim, J. Chung, I.C. Choi, A comparison study of cluster validity indices using a nonhierarchical clustering algorithm, in: Proceedings of CIMCA-IAWTIC'05, 01, 2005, pp. 199–204.
[16] S. Ray, R.H. Turi, Determination of number of clusters in k-means clustering and application in colour image segmentation, in: Proceedings of the 4th International Conference on Advances in Pattern Recognition and Digital Techniques (ICAPRDT'99), 1999, pp. 137–143.
[17] D.L. Davies, D.W. Bouldin, A cluster separation measure, IEEE Transactions on Pattern Analysis and Machine Intelligence 1 (2) (1979) 224–227.
[18] D.B. Rubin, Inference and missing data, Biometrika 63 (3) (1976) 581.
[19] R.J. Little, Modeling the drop-out mechanism in repeated-measures studies, Journal of the American Statistical Association 90 (431) (1995) 1112–1121.
[20] J.M. Engels, P. Diehr, Imputation of missing longitudinal data: a comparison of methods, Journal of Clinical Epidemiology 56 (10) (2003) 968–976.
[21] C.M. Genolini, KmL: K-Means for Longitudinal Data (KmL), R package version 1.1.3, http://CRAN.R-project.org/package=kml
[22] P. Jaccard, Distribution de la flore alpine dans le Bassin des Drouces et dans quelques régions voisines, Bulletin de la Société Vaudoise des Sciences Naturelles 37 (140) (1901) 241–272.
[23] T.P. Beauchaine, R.J. Beauchaine, A comparison of maximum covariance and K-means cluster analysis in classifying cases into known Taxon Groups, Psychological Methods 7 (2) (2002) 245–261.
[24] J.B. Pingault, R.E. Tremblay, F. Vitaro, R. Carbonneau, C.M. Genolini, B. Falissard, S.M. Côté, Childhood trajectories of inattention and hyperactivity and prediction of educational attainment in early adulthood: a 16-year longitudinal population-based study, American Journal of Psychiatry 168 (11) (2011) 1164–1170.
[25] S.S. Lee, S.P. Hinshaw, Predictors of adolescent functioning in girls with attention deficit hyperactivity disorder (ADHD): the role of childhood ADHD, conduct problems, and peer status, Journal of Clinical Child and Adolescent Psychology 35 (3) (2006) 356–368.
[26] G.M. Massetti, B.B. Lahey, W.E. Pelham, J. Loney, A. Ehrhardt, S.S. Lee, H. Kipp, Academic achievement over 8 years among children who met modified criteria for attention-deficit/hyperactivity disorder at 4–6 years of age, Journal of Abnormal Child Psychology 36 (3) (2008) 399–410.
[27] R.E. Tremblay, R. Loeber, C. Gagnon, P. Charlebois, S. Larivée, M. LeBlanc, Disruptive boys with stable and unstable high fighting behavior patterns during junior elementary school, Journal of Abnormal Child Psychology 19 (3) (1991) 285–300.
[28] E. Touchette, S.M. Côté, D. Petit, X. Liu, M. Boivin, B. Falissard, R.E. Tremblay, J.Y. Montplaisir, Short nighttime sleep-duration and hyperactivity trajectories in early childhood 124 (5) (2009) e985–e993.
[29] P. Cummings, The relative merits of risk ratios and odds ratios, Archives of Pediatrics & Adolescent Medicine 163 (May) (2009) 438–445.